

# A new algorithm of Langevin simulation and its application to the SU(2) and SU(3) lattice gauge \*

Hideo Nakajima<sup>a †</sup> and Sadataka Furui<sup>b ‡</sup>

<sup>a</sup>Department of Information Science, Utsunomiya University,  
2753 Ishii, Utsunomiya 321 Japan

<sup>b</sup>School of Science and Engineering, Teikyo University,  
1-1 Toyosatodai, Utsunomiya 320, Japan

The 2nd order Runge-Kutta scheme Langevin simulation of unquenched QCD in pseudofermion method derived from our general theory shows a behaviour as a function of the Langevin step  $t$  better than the Fukugita, Oyanagi, Ukawa's scheme.

## 1. Introduction

In order to incorporate fermions in the lattice QCD, Hybrid Monte Carlo (HMC) is the standard method[1], but in the case of large lattice, the method is not free from problems caused by rounding errors[2]. The Langevin simulation is another possible method to simulate fields including fermions with or without gauge fixing.

The discretized Langevin simulation results depend on the step size  $t$ , and the incorporation of higher order corrections in the step function to reproduce the Fokker-Planck distribution is the problem. Drummond et al[3] showed a method to derive higher order corrections up to second order in the step in U(1) spin model and in SU(N) lattice gauge theory. We obtained the Langevin step function up to third order in flat space and up to second order in curved space[4].

A Langevin simulation using the Runge-Kutta type algorithm including dynamical quark loops and corrections up to second order in  $t$  was proposed by Batrouni et al[5] and, Ukawa and Fukugita[6].

An extensive full-QCD Langevin simulation was performed by Fukugita, Oyanagi and Ukawa[7] in partial 2nd order bilinear noise scheme. They reported that the 2nd order

pseudo-fermion scheme is worse than the partial 2nd order bilinear noise scheme in a simulation of  $4^3 \times 8$  lattice.

In this work, we extend the new algorithm of Langevin simulation to an algorithm in the canonical coordinate space of Lie algebra of SU(N). We then apply the method to SU(N) lattice gauge theory including pseudofermion and discuss possible improvements.

## 2. Random walks in the space of Lie groups

In the space of SU(N) Lie groups, a group element  $g$  is expressed as  $g = \exp(x \cdot \lambda)$  where  $x_i$ 's are the canonical coordinates of the first kind, and  $\lambda_i$ 's are antihermitean matrices which span space of  $su(N)$  Lie algebra as  $[\lambda_i, \lambda_j] = c_{ijk} \lambda_k$  and are normalized as  $Tr(\lambda_i^\dagger \lambda_j) = \delta_{ij}$ .

The right differentiation  $\nabla$  is defined for any  $f(g)$  as  $e^{X \cdot \nabla} f(g) = f(ge^{X \cdot \lambda})$ , where  $\nabla_i$ 's follow the same commutation relation as  $\lambda_i$ 's and

$$e^{-X \cdot \nabla} \delta^{inv}(g - g_0) = \delta^{inv}(g - g_0 e^{X \cdot \lambda}) \quad (1)$$

The Fokker-Planck equation in the space of Lie group is  $\partial_t \phi = K \phi$ , where  $K = \nabla_k (\nabla_k - u_k)$ , and  $\phi$  is normalized with the Haar measure.

If  $u_i = \nabla_i S$ , then  $\phi$  tends to its asymptotic distribution  $const \cdot e^S$ . The formal solution  $\phi$  with the initial condition  $\phi(g) = \delta^{inv}(g - g_0)$  is

$$\phi = e^{tK} \delta^{inv}(g - g_0). \quad (2)$$

\*Presented by S. Furui

<sup>†</sup>e-mail nakajima@kinu.infor.utsunomiya-u.ac.jp

<sup>‡</sup>e-mail furui@dream.ics.teikyo-u.ac.jp

The Langevin simulation algorithm  $X_i$  for the random walk in the Lie algebra space is defined as

$$e^{tK} \delta^{inv}(g - g_0) = \langle \delta^{inv}(g - g_0 e^{X \cdot \lambda}) \rangle, \quad (3)$$

where  $X$  is a function determined from  $s = \sqrt{t}$ , and all values of  $u_i$  together with their higher derivatives evaluated at  $g_0$ , and independent Gaussian random variates  $\xi$ 's with variance 2 and mean 0, and  $\langle \dots \rangle$  denotes an averaging over the Gaussian distribution.

Thus the problem is to find the  $X_i$  such that

$$\langle g | \exp(t\hat{K}) | g_0 \rangle = \langle g | \langle e^{-X(s, \xi) \cdot \nabla} | g_0 \rangle \rangle \quad (4)$$

where  $\hat{K} = \hat{\nabla} \cdot (\hat{\nabla} - u(\hat{g}))$ , and note that operators are normal ordered, i.e., all  $\hat{\nabla}$ 's are set in the leftmost position relative to all  $\hat{g}$ 's on the l.h.s. of the equation.

In order to get  $X$ , let  $X_i$  be expanded as

$$X_i = \sum_{n=0}^{\infty} \sum_{k=0}^n s^n f_{(n,k)} h_k(\xi) \quad (5)$$

where  $h_k(\xi)$  is a generalized  $k$ -th order Hermite polynomial. First, put  $\exp(t\hat{K})$  in normal order as mentioned above. Then starting from  $X_i = s\xi_i + \dots$ , and considering operator equations for each order  $O(s^n)$ , and then for given  $n$ ,  $O(\nabla^m)$  in larger- $m$ -first order, we proceed perturbatively and obtain  $f_{(n,k)}$ 's. We can prove this procedure works satisfactorily, i.e., a newly fixed  $f_{(n,k)}$  does not affect lower order equations.

Our  $O(s^6)$  algorithm is as follows,

$$\begin{aligned} X_i = & s\xi_i + s^2 u_i + \frac{s^3}{12} (6\xi_j \nabla_j u_i - N\xi_i) \\ & + \frac{s^4}{6} (Nu_i + 3u_j \nabla_j u_i + \nabla^2 u_i + \xi_j \xi_k \nabla_j \nabla_k u_i) \\ & + \frac{s^5}{288} (12c_{ijk} c_{klm} \xi_l u_j u_m - 36N\xi_j \nabla_j u_i \\ & + 12\xi_j \nabla_k u_j \nabla_k u_i + 48\xi_j \nabla_j u_k \nabla_k u_i \\ & + 48\xi_j u_k \nabla_k \nabla_j u_i + 48\xi_j u_k \nabla_j \nabla_k u_i \\ & + 96\xi_j \nabla_j \nabla^2 u_i + 24c_{ijk} c_{ljm} \xi_l \nabla_k u_m \\ & + 24c_{ijk} \xi_l u_j \nabla_l u_k + 48c_{ijk} \xi_l \nabla_j \nabla_l u_k \\ & - 24c_{jkl} \xi_j u_k \nabla_l u_i + 48c_{jkl} \xi_j \nabla_k \nabla_l u_i - N^2 \xi_i) \\ & + \frac{s^6}{60} (-30Nu_j \nabla_j u_i - 10\nabla^2 u_j \nabla_j u_i \end{aligned}$$

$$\begin{aligned} & -10u_j \nabla_j u_k \nabla_k u_i - 30N\nabla^2 u_i \\ & -20\nabla_j u_k \nabla_j \nabla_k u_i - 10u_j u_k \nabla_k \nabla_j u_i \\ & -20u_j \nabla_j \nabla^2 u_i - 10\nabla^2 \nabla^2 u_i - 10c_{ijk} \nabla_j u_l \nabla_l u_k \\ & -5c_{ijk} u_j \nabla^2 u_k - 5c_{ijk} u_j u_l \nabla_l u_k \\ & -5c_{ijk} c_{ljm} u_l \nabla_m u_k - 5c_{ijk} c_{lkm} u_l \nabla_j u_m \\ & -54N^2 u_i) \end{aligned} \quad (6)$$

In the case of  $SU(3)$ , exponentiation of the matrix  $x \cdot \lambda$  is done analytically as follows.

$$e^{x \cdot \lambda} = e^{iz_1} P_1 + e^{iz_2} P_2 + e^{iz_3} P_3. \quad (7)$$

with  $z_i$  as the eigenvalue given by the Cardano method, and  $P_i$  as the projection operator on its eigenspace. The unitarity of the matrix is preserved in very high accuracy.

### 2.1. QCD with pseudofermion

In order to incorporate quark loops in the vacuum, we include in addition to the gauge field  $U$ , a pseudofermion field  $\phi$  which is a complex-scalar, and define the partition  $Z = \int d\phi^* d\phi dU e^{\beta S}$ .

The action is

$$S = S_{gauge} + S_{fermi} \quad (8)$$

where

$$S_{gauge} = \frac{1}{N} \sum_p \text{Re}(\text{Tr} U_p) \quad (9)$$

The matrix elements for Wilson fermions are calculated by using hermitian  $\gamma$  matrices and the hopping parameter  $\kappa$ .

After the preconditionings the fermionic action can be replaced by the even-site fermionic action[10]

$$S_{fermi} = -\phi_e^* \frac{1}{\tilde{M}\tilde{M}^\dagger} \phi_e \quad (10)$$

where  $\tilde{M} = 1 - \kappa^2 DD$  and  $D_{xy} = \sum_\mu \{ (1 - \gamma_\mu) U_{x,\mu} \delta_{x+\mu,y} + (1 + \gamma_\mu) U_{y,\mu}^\dagger \delta_{x-\mu,y} \}$ .

The  $O(s^4)$  Runge-Kutta algorithm  $X$  for the  $SU(N)$  link variables  $U$  is

$$X = \frac{N}{12} s^2 (-s\xi + s^2 2u) + s\xi + \frac{1}{2} s^2 (u + \tilde{u}) \quad (11)$$

where  $\tilde{u} = u(g e^{(s\xi + s^2 u) \cdot \lambda})$ . The velocity field is

$$u_{i\mu} = \nabla_\mu^i S_{gauge} + \nabla_\mu^i S_{fermi}. \quad (12)$$

where,  $\nabla_\mu^i$  is the right differentiation with respect to the  $i$ -th canonical coordinate of  $U_{x,\mu}$ .

The function  $X$  for the pseudofermion is similar but the  $N$  dependent term is absent,  $u_\phi = -2\frac{1}{MM^\dagger}\phi$  and  $u_{\phi^*} = -2\phi^*\frac{1}{MM^\dagger}$ .

### 3. Numerical Results

In the second order Taylor scheme Langevin simulation of SU(2) on  $8^4$  lattice, we found the dependence on  $t$  of the data simulated in the space of Lie algebra is much better than that in the space of  $S^3$ [8].

Simulations of SU(2) single degree of freedom, and of  $4^4$  and  $8^4$  lattices were performed in the Taylor scheme and in the Runge-Kutta scheme. In the single degree of freedom both showed similar dependence, however in the  $4^4$  lattice, the behaviour of the Taylor scheme was worse than the Runge-Kutta scheme at large  $t$ [8].

In the case of SU(2) single degree of freedom, the third order Taylor scheme did not improve the second order scheme, but Runge-Kutta scheme that incorporates the third order term is expected to improve the dependence on  $t$  in the lattice simulation.

We checked the results of SU(3) by comparing with the Monte Carlo simulation, in the Cabibbo-Marinari[11]'s scheme. The dependence of  $\langle U_P \rangle = \langle \text{Re}(\frac{1}{3}\text{Tr}U_P) \rangle$  on  $\kappa$  in the  $4^4$  lattice in the 2nd order Langevin simulation of full-QCD with pseudofermion is shown in Fig.1. At  $t = 0.05$ , the agreement to the HMC result is better than that of Fukugita, Oyanagi and Ukawa[7] and at  $t = 0.01$  they are consistent within statistical errors.

### 4. Outlook

General algorithm for obtaining higher order weak Taylor approximations to the Langevin step algorithms with respect to  $t$ , improved version in practical use for a finite fixed order was formulated. (cf. Nakajima [9] for full order weak Taylor algorithm in recursion formula)

We try to obtain a local  $O(t^3)$  Runge-Kutta like algorithm in SU(3) and aim to use it for investigation of its performance in full QCD simulation.

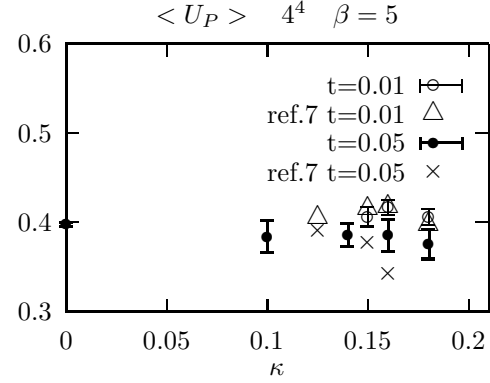


Figure 1. Dependence of Wilson loop  $U_P = \text{Re}(\frac{1}{3}\text{Tr}U_P)$  on the hopping parameter  $\kappa$ . The triangle at  $\kappa = 0.15$  is the result of HMC[1].

This work was supported by National Laboratory for High Energy Physics, as KEK Supercomputer Project ( Project No.97-23 ).

### REFERENCES

1. R.Gupta et al., Phys. Rev. **D40** (1989) 2072.
2. K.Jansen, Nucl. Phys **B** (Proc. Suppl.) 53,(1997) 127 .
3. I.T.Drummond, S.Duane and R.R.Horgan, Nucl. Phys **B220** [FS8] (1983) 119.
4. H.Nakajima and S.Furui, Nucl. Phys **B** (Proc. Suppl.) 53, (1997) 983.
5. G.G.Batrouni et al., Phys. Rev.**D32** (1985) 2736;
6. A.Ukawa and M.Fukugita, Phys Rev. Lett.**55**(1985) 1854 .
7. M.Fukugita, Y.Oyanagi and A.Ukawa, Phys. Rev.**D36**(1987) 824.
8. H.Nakajima and S. Furui, Soryushiron Kenkyuu,**95** (1997) D217, YKIS'96 proc.
9. H.Nakajima, Utsunomiya University preprint UTUP-130, hep-lat9703013
10. T.A.Degrand and P.Rossi, Comp. Phys. Comm. **60** (1990) 211.
11. N.Cabibbo and E.Marinari, Phys. Lett. **119B** (1982) 387.